CLAIMS

1. A method of calculating parity segments comprising:

providing a parity calculation module configured to calculate one or more parity segments, the parity calculation module being embodied as an application-specific integrated circuit (ASIC);

with the ASIC:

receiving one or more data segments that are to be used to calculate one or more parity segments;

receiving one or more parity coefficients that are to be used to calculate the one or more parity segments, wherein:

the one or more parity coefficients are chosen from a plurality of coefficient subsets; and

each said coefficient subset is classified based on a respective parity operation into one of a plurality of groups;

operating on the one or more data segments and the one or more parity coefficients to provide an intermediate computation result;

writing the intermediate computation result to one or more local buffers on the ASIC; and

using the intermediate computation result from the one or more local buffers to calculate one or more parity segments.

2. The method of claim 1, wherein the ASIC has multiple local memory components to hold data that is used in the calculation of the parity segments.

3. The method of claim 1, wherein said act of operating is performed by one or more finite mathematical operator components.

4. The method of claim 1 further comprising maintaining multiple parity coefficients in one or more local memory components on the ASIC thereby reducing external memory access operations.

5. The method of claim 4, wherein said receiving one or more parity coefficients comprises receiving the coefficients from the one or more local memory components and into one or more finite mathematical operator components that are configured to provide the intermediate computation result.

6. The method of claim 1 further comprising providing feedback from the one or more local buffers to one or more mathematical operator components that are configured to perform said operating.

7. The method of claim 6 further comprising:

receiving one or more additional data segments that are to be used to calculate one or more parity segments;

receiving one or more additional parity coefficients that are to be used to calculate the one or more parity segments;

receiving the intermediate computation result from the one or more local buffers;

operating on the one or more additional data segments, the one or more additional parity coefficients, and the intermediate computation result to provide a result; and

writing the result to one or more local buffers on the ASIC.

8. The method of claim 7, wherein said result that is provided by said operating on the one or more additional data segments, the one or more additional parity coefficients, and the intermediate computation result comprises an additional intermediate computation result.

9. The method of claim 7, wherein said result that is provided by said operating on the one or more additional data segments, the one or more additional parity coefficients, and the intermediate computation result comprises one or more parity segments.

10. The method of claim 7, wherein said one or more local buffers comprise SRAMs.

11. The method of claim 7, wherein said one or more local buffers comprise SRAMs, and said acts of claim 7 are performed within one clock cycle of a system clock.

12. The method of claim 1, wherein said one or more local buffers comprise SRAMs.

13. A method of calculating parity segments comprising:

providing a parity calculation module configured to calculate one or more parity segments, the parity calculation module being embodied as an application-specific integrated circuit (ASIC);

with the ASIC:

receiving one or more data segments that are to be used to calculate one or more parity segments;

receiving one or more parity coefficients that are to be used to calculate the one or more parity segments;

operating on the one or more data segments and the one or more parity coefficients to provide an intermediate computation result;

writing the intermediate computation result to one or more local buffers on the ASIC;

using the intermediate computation result from the one or more local buffers to calculate one or more parity segments; and

providing feedback from the one or more local buffers to one or more mathematical operator components that are configured to perform said operating, wherein said feedback on a first pass through the one or more mathematical operator components does not affect computations performed by the one or more mathematical operator components.

14. The method of claim 13, wherein said feedback on the first pass is zeroed out.

15. A method of calculating parity segments comprising:

providing a parity calculation module configured to calculate one or more parity segments;

with the parity calculation module:

receiving one or more data segments that are to be used to calculate one or more parity segments;

receiving one or more parity coefficients that are to be used to calculate the one or more parity segments;

operating on the one or more data segments and the one or more parity coefficients to provide an intermediate computation result;

writing the intermediate computation result to one or more local buffers; and

within one clock cycle of an associated clock, receiving (a) the intermediate computation result from the one or more local buffers, (b) one or more additional data segments and (c) one or more additional parity coefficients, and operating on them to provide a result that is stored in the one or more local buffers.

16. The method of claim 15, wherein the parity calculation module comprises an application specific integrated circuit (ASIC).

17. The method of claim 15, wherein the one or more local buffers comprise SRAMs.

18. The method of claim 15, wherein the parity calculation module comprises an application specific integrated circuit (ASIC), and the one or more local buffers comprise SRAMs on the ASIC.

19. The method of claim 15, wherein the parity calculation module comprises one or more local memory components configured to locally hold data that is used in the calculation of the parity segments.

20. A parity segment calculation module comprising:

an application specific integrated circuit (ASIC) having at least:

one or more result buffers for holding intermediate computation results;

one or more mathematical operator components configured to receive data segments and coefficients associated with the data segments and operate on them to provide intermediate computation results that can be written to the one or more result buffers, wherein the coefficients are chosen from a plurality of coefficient subsets, each said coefficient subset is classified based on a respective parity operation; and

one or more feedback lines, individual lines being coupled between an associated result buffer and an associated mathematical operator component, to provide an intermediate computation result to the math operator for use in calculating parity segments.

21. The parity segment calculation module of claim 20, wherein the one or more result buffers comprise at least one SRAM.

22. The parity segment calculation module of claim 20, wherein the one or more result buffers comprise multiple SRAMs.

23. The parity segment calculation module of claim 20, wherein the one or more result buffers comprise two SRAMs.

24. A method of calculating parity segments comprising:

providing a parity calculation module configured to calculate one or more parity segments;

with the parity module:

receiving one or more data segments that are to be used to calculate one or more parity segments;

receiving one or more parity coefficients that are to be used to calculate the one or more parity segments, wherein:

the one or more parity coefficients are chosen from a plurality of coefficient subsets; and

each said coefficient subset is classified based on a respective parity operation into one of a plurality of groups;

operating on the one or more data segments and the one or more parity coefficients to provide an intermediate computation result;

writing the intermediate computation result to one or more local buffers; and

using the intermediate computation result from the one or more local buffers to calculate one or more parity segments.

25. The method of claim 24, wherein the parity module has multiple local memory components to hold data that is used in the calculation of the parity segments.

26. The method of claim 24, wherein said act of operating is performed by one or more finite mathematical operator components.

27. The method of claim 24 further comprising maintaining multiple parity coefficients in one or more local memory components on the parity module thereby reducing external memory access operations.

28. The method of claim 27, wherein said receiving one or more parity coefficients comprises receiving the coefficients from the one or more local memory components and into one or more finite mathematical operator components that are configured to provide the intermediate computation result.

29. The method of claim 24 further comprising providing feedback from the one or more local buffers to one or more mathematical operator components that are configured to perform said operating.

30. The method of claim 29 further comprising:

receiving one or more additional data segments that are to be used to calculate one or more parity segments;

receiving one or more additional parity coefficients that are to be used to calculate the one or more parity segments;

receiving the intermediate computation result from the one or more local buffers;

operating on the one or more additional data segments, the one or more additional parity coefficients, and the intermediate computation result to provide a result; and

writing the result to one or more local buffers on the parity module.

31. The method of claim 30, wherein said result that is provided by said operating on the one or more additional data segments, the one or more additional parity coefficients, and the intermediate computation result comprises an additional intermediate computation result.

32. The method of claim 30, wherein said result that is provided by said operating on the one or more additional data segments, the one or more additional parity coefficients, and the intermediate computation result comprises one or more parity segments.

33. The method of claim 30, wherein said one or more local buffers comprise SRAMs.

34. The method of claim 30, wherein said one or more local buffers comprise SRAMs, and said acts of claim 30 are performed within one clock cycle of a system clock.

35. The method of claim 24, wherein said one or more local buffers comprise SRAMs.